等号を含む第一階時相論理のサブクラスとその恒真性判定問題(計算モデルと計算の複雑さに関する研究)

AUTHOR(S):

浜口, 清治; 矢島, 脩三

CITATION:

浜口, 清治 ...[et al]. 等号を含む第一階時相論理のサブクラスとその恒真性判定問題(計算モデルと計算の複雑さに関する研究). 数理解析研究所講究録 1996, 950: 193-199

ISSUE DATE:

1996-05

URL:

http://hdl.handle.net/2433/60320

数理解析研究所講究録
950巻 1996年 193-199

等号を含む第一階時相論理のサブクラスとその恒真性判定問題

浜口清治　　矢島脩三
Kiyoharu HAMAGUCHI　Shuzo YAJIMA
京都大学大学院工学研究科情報工学専攻

等号を含み、限量子を含まない第一階述語論理の恒真性判定問題は決定可能であるあることが知られているが、時相演算子を付け加えて拡張しても、決定可能であることを示す。

1　はじめに

命題論理式の恒真性判定は組合せ回路の等価性判定に応用することができることから、実用を目指した研究が多数なされてきている。例えば、論理関数を表現するためのデータ構造である二分決定グラフ [1] を用いることによって、近年では実用規模の組合せ回路の等価性判定が可能となり、形式的検証の一手法として用いられるようになってきている。また、命題論理に時間に係わる性質を記述するための時相演算子を加えた命題時相論理は、順序回路などの有限状態機械に対する仕様または部分仕様の記述に用いられており、二分決定グラフを利用する手法によって、かなり大規模の回路を扱うことが可能になっている。

しかし、特に順序回路に関しては、大規模な実用回路を取り扱うことは困難であると考えられている。例えば、順序回路を検証する場合、状態変数の数が、50 ないし 60 を越えるとほとんどの場合取り扱うことができなくなる。このため、順序回路内に含まれるデータパスを抽象化して、形式的設計検証を行う試みがある。これは、従来論理回路レベルで行われていた検証をレジスタ転送レベル以上のレベルで行うことを意味する。この手法では、算術演算などを記号のまま操作するために、命題論理よりも強力な論理体系を用いることになる。

第一階述語論理における充足可能性判定問題および恒真性判定問題は、一般に決定不能であるが、$\forall x_1 \forall x_2 \cdots \forall x_n A$ (A は限量子を含まない論理式) の形の論理式の恒真性判定問題は、述語として等号を加えた場合でも、決定可能であることが示されている [2]。Jones らは、この点に注目して、等号を含む第一階述語論理のサブクラスを用いてマイクロプロセッサの検証を行っている [3]。問題が決定可能なので、完全な自動化が可能になるという特長がある。しかし、この論理では命題時相論理の場合のように、時間に関する性質を扱うことができない。

本稿では、上述の等号を含む第一階述語論理のサブクラスに時相演算子を導入した体系とその恒真性判定アルゴリズムを示す。このアルゴリズムは線形時相論理の充足可能性判定で用いられるタブロー法 [4, 5] と等号を含む第一階述語論理に対する充足可能性判定で用いられる合同閉包を利用したものである。以下、等号を含む第一階述語論理のサブクラスとその決定手続きを紹介して、その後、これに時相演算子を導入する。

2　等号を含む第一階述語論理のサブクラスとその決定手続き

2.1　等号を含む第一階述語論理のサブクラス

まず、等号を含む第一階述語論理のサブクラスを示す。限量子は用いられないが、意味的には各変数は全称限量子で束縛されていることと等価になる。このような制限のもとでは論理式の恒真性判定問題が決定可能となる。

定義 1 構文
項は再帰的に次のように定義される。

- 変数 $x, y, z, \cdots$ は項。
- 定数 $a, b, c, \cdots$ は項。
- f が n 個の引数を持つ関数で、$t_1, t_2, \cdots, t_n$ が項ならば、$f(t_1, t_2, \cdots, t_n)$ も項である。

論理式は次のように再帰的に定義される。

- P が引数 n 個の述語で、$t_1, t_2, \cdots, t_n$ が項ならば、$P(t_1, \cdots, t_n)$ は論理式である。
- 引数 2 個の特別な述語 $\doteq$ に対して、t と u が項ならば、$t \doteq u$ は論理式である。
- A と B が論理式の時、$A \vee B$ と $\neg A$ も論理式である。

$\wedge$(論理積), $\Rightarrow$(含意) などの演算子も、通常通り、$\vee$(論理和) と $\neg$ (否定) によって定義することができる。以下本稿でもこれらを用いる。$\doteq$, 二項演算子、単項演算子の順で優先順位が高いものとする。 □

[例] $((y \doteq f(x)) \wedge (w \doteq g(y,z))) \Rightarrow (w \doteq g(f(x),z))$ □

定義 2 意味
解釈
解釈 I は、領域と呼ばれる集合 U と各記号の解釈 $\langle\ \rangle$ からなる。U はブール値 $\{true, false\}$ を含むものとする。

- c が定数の時、値 $\langle c \rangle \in U$
- f が関数記号の時、関数 $\langle f \rangle : U^n \to U$
- P が n 引数の述語記号の時、関数 $\langle P \rangle : U^n \to \{true, false\}$

値の評価
解釈 I および、変数集合 $V = \{x_1, x_2, \cdots, x_n\}$ の各変数 x_i への U の要素の割当てを行う関数 $s : V \to U$ が与えられているとする。この時、項 t および 論理式 A の I と s のもとでの値の評価 $t_{[I,s]}$ と $A_{[I,s]}$ は次のように再帰的に定義される。

- t が定数の時、$t_{[I,s]} = \langle t \rangle$
- t が変数の時、$t_{[I,s]} = s(t)$
- t が $f(t_1, t_2, \cdots, t_n)$ の形の項の時、$t_{[I,s]} = \langle f \rangle (t_{1\,[I,s]}, \cdots, t_{n\,[I,s]})$
- A が $P(t_1, \cdots, t_n)$ の論理式であり、P が $\doteq$ とは異なる時、$A_{[I,s]} = \langle A \rangle (t_{1\,[I,s]}, \cdots, t_{n\,[I,s]})$
- A が $t \doteq u$ の時、$t_{[I,s]} = u_{[I,s]}$ ならば $true$, そうでなければ $false$.
- A が $B \vee C$ および $\neg B$ の時、$A_{[I,s]}$ はそれぞれ $B_{[I,s]}$ と $C_{[I,s]}$ の論理積および $B_{[I,s]}$ の否定。

恒真性と充足可能性
論理式 A が任意の解釈 I と変数への値の割り当て s のもとで、$A_{[I,s]} = true$ となる時、論理式 A は恒真であるといい、$\models A$ と書く。また、ある解釈 I と変数への値の割り当て s のもとで、$A_{[I,s]} = true$ となる時、論理式 A は充足可能であるといい、そうでない時、充足不能であるという。 □

論理式 A の恒真性判定問題は論理式 $\neg A$ の充足可能性判定問題に帰着できる。この問題は決定可能であることが知られている。以下では、Oppen と Nelson のアルゴリズムに基づく手法を示す [2, 6]。
[例] $B = (y \doteq f(x) \wedge w \doteq g(y,z)) \Rightarrow w \doteq g(f(x),z)$ は恒真であり、$\neg B$ は充足不能である。□

2.2 充足可能性判定アルゴリズム

定義 3 論理式 A が与えられた時、$D = \{t \doteq u \mid t$ と u は A 中に出現する項$\}$ を考える。$C = C_1 \vee C_1 \vee \cdots \vee C_n$ が A の完全な積和形であるとは、$A = C$ かつ $i \neq j$ ならば $C_i \neq C_j$、また各 C_i には D の任意の要素が肯定または否定形で出現することをいう。 □

この変形は D の要素を原子命題とみなして、A を積和標準形に変形する操作に等しい。

アルゴリズム 1 充足可能性判定アルゴリズム
論理式 A の充足可能性判定は次の 3 ステップからなる。

1. A 中に出現する述語 $P(t_1,\cdots,t_n)$ を $P(t_1,\cdots,t_n) \doteq T$ によって置き換える。ここで、T は特別な定数記号である。以下、P を関数記号と見なして、$P(t_1,\cdots,t_n)$ を項として取り扱う。
2. A を完全な積和形 $C_1 \vee C_1 \vee \cdots \vee C_n$ に変形する。
3. 各積項 C_i について充足可能性を判定する。手法は以下に述べる。充足可能な積項があれば、論理式 A も充足可能である。

□

1. と 2. により得られた各積項 C_i は一般に次の形になる。

$$t_1 \doteq u_1 \wedge \cdots \wedge t_n \doteq u_n \wedge \neg v_1 \doteq w_1 \wedge \cdots \wedge \neg v_p \doteq w_p$$

充足不能性を判定するために、式中に含まれる項全ての解析木に対応するグラフ $G(V,\Lambda,\delta)$ を導入する。

- V は節点の集合。
- $v \in V$ に対して、$\Lambda(v)$ は対応する項の中の記号を与える。
- $\delta(v,i)$ は節点 v に対応する記号の i 番めの引数に対応する節点を与える。対応する引数がなければ $\perp$ を返す。

定義 4 積項 C に対して定まるグラフ $G(C)$ について、次の条件を満足する関係 $R \subseteq V \times V$ を $G(C)$- 合同 ($G(C)$-congruent) な関係 (または単に合同な関係) であるという。

- $\Lambda(u) = \Lambda(v)$ かつ 全ての i について、$\delta(u,i)\ R\ \delta(v,i)$ ならば uRv。
- R は同値関係。

関係 E を含む最も細かい合同な関係 R のことを E の合同閉包 (congruence closure) という。ここでいう最も細かい合同な関係 R とは、R による同値類のいずれかを分割すると、R が上記の条件を満足する合同な関係でなくなることをいう。 □

[例] 積項 $(y \doteq f(x)) \wedge (w \doteq g(y,z)) \wedge \neg(w \doteq g(f(x),z))$ に対する $E = \{(y,f(x)),(w,g(y,z))\}$ の合同閉包 R は $\{(y,f(x)),(w,g(y,z)),(g(y,z),g(f(x),z)),(w,g(f(x),z))\}$ となる。 □

次の結果が知られている [6]。

命題 1 積項の充足可能性判定
積項

$$C = t_1 \doteq u_1 \wedge \cdots \wedge t_n \doteq u_n \wedge \neg v_1 \doteq w_1 \wedge \cdots \wedge \neg v_p \doteq w_p$$

について、$E = \{(t_1,u_1),\cdots,(t_n,u_n)\}$ の合同閉包 $\overset{*}{\leftrightarrow}$ を求める。この時、

積項 C が充足可能 $\Leftrightarrow$ どの j $(j = 1,\cdots,p)$ についても $v_j \overset{*}{\leftrightarrow} w_j$ ではない。

□

E 中に出現する項とその部分項の数を m とすると合同閉包は $O(m \log m)$ で求めることができることが知られている [7]。

上記の例を見てみると、$y \doteq f(x)$ かつ $w \doteq g(y,z)$ であることから、合同閉包に $(w,g(f(x),z))$ が含まれるが、この積項には $\neg(w \doteq g(f(x),z))$ が含まれているため、充足不可能であることがわかる。

論理式 A が充足可能な場合には、その完全な積和形中に充足可能な積項 C が存在する。C から導出される合同閉包に基づいて、A を充足する解釈と値の割り当てを作ることができる。直観的には、合同閉包中の同値類 e_i $(i = 1,2,\cdots,n)$ について代表元からなる集合を領域 U として、e_i に含まれる各項を e_i の代表元に写像するように、解釈 I と値の割り当て s を定める。

アルゴリズム 2 論理式を充足する解釈と値の割り当て

論理式 A が充足可能で、その完全な積和形中の積項 C が充足可能であるとする。以下では、C 中に出現する項 t については、$[t]$ は、t の同値類の代表元を表すものとする。論理式 A 中には出現するが、C 中には出現しない項 t については、$[t] = \perp$ とする。

- 領域 U : 合同閉包中の同値類の代表元、*true*, *false*, $\perp$ からなる集合。
- 各記号の解釈
  - 定数 c : C 中に出現する場合 $\langle c \rangle = [c]$. C 中に出現しない場合、$\langle c \rangle = \perp$.
  - 関数 f : C 中に出現する $f(t_1, \cdots, t_n)$ については、$[f(t_1, \cdots, t_n)] = \langle f \rangle([t_1], \cdots, [t_n])$ となるように、それ以外の場合は $\perp = \langle f \rangle([t_1], \cdots, [t_n])$ となるように $\langle f \rangle$ を定める。
  - 述語記号 P : C 中に出現する $P(t_1, \cdots, t_n)$ については、$[P(t_1, \cdots, t_n)] = [T]$ ならば、$true = \langle P \rangle([t_1], \cdots, [t_n])$ となるように、それ以外の場合は $false = \langle P \rangle([t_1], \cdots, [t_n])$ となるように $\langle P \rangle$ を定める。
- 変数への値の割り当て s : 各変数 x が C 中に出現する場合 $s(x) = [x]$、それ以外の場合 $s(x) = \perp$ となるように s を定める。

3 等号を含む第一階時相論理のサブクラスとその恒真性判定

3.1 等号を含む第一階時相論理のサブクラス

時相演算子 $\mathbf{X}$ と $\mathcal{U}$ を次のように導入する。

定義 5 構文

関数記号を静的関数記号と動的関数記号に区別する。
項について次の定義を加える。

- t が項ならば、$\mathbf{X}t$ も項である。

論理式について次の定義を加える。

- A と B が論理式ならば $\mathbf{X}A$ および $A \,\mathcal{U}\, B$ も論理式である。

□

t が変数の場合は $\mathbf{X}t$ は次の時刻での t の値を意味する。$\mathbf{X}f(t_1, t_2, \cdots, t_n)$ の場合は、現時刻の $t_1, t_2, \cdots, t_n$ の値を入れたときの次の時刻での f を評価した値を意味する。$\mathbf{X}f(\mathbf{X}t_1, \cdots, \mathbf{X}t_n)$ の場合は、次時刻の $t_1, t_2, \cdots, t_n$ の値を入れた時の次の時刻での f を評価した値を意味する。演算子 $\mathbf{X}$ は直後の記号についてのみ、次の時刻での解釈または値を使って評価することを意味している。

A が論理式の場合は、$\mathbf{X}A$ は次の時刻で A が成立することを意味し、$A \,\mathcal{U}\, B$ は B が成り立つまで A が成立することを意味する。

[例] $\neg(x \doteq \mathbf{X}y) \,\mathcal{U}\, (x \doteq \mathbf{X}y)$ □

論理式 A の恒真性判定では、各記号に対する解釈と変数への値の割当てをすべて考える。この問題はある記号の解釈と変数への割当てで $\neg A$ を真にするものがあるかどうかを求める問題である。この問題では、変数を特に区別する必要がないので、定数、変数を 0 引数の関数記号とみなすことにする。また、変数への値の割り当て s は考えないことにする。

定義 6 意味

論理式 A の値の評価は、解釈の無限系列 $\mathcal{I} = I_1, I_2, \cdots,$ に対して行われる。$\mathcal{I}^j = I_j, I_{j+1}, \cdots,$ と定める。以下では、f_{I_j} および P_{I_j} は解釈 I_j における t と P をそれぞれ意味する。

$\mathcal{I}$ には次の制約を設ける。

- 静的関数記号についてはその解釈は全ての I_i に対して同一である。

$\mathcal{I}$ における項 t と論理式 A の値 $t_{\mathcal{I}}$ および $A_{\mathcal{I}}$ はそれぞれ再帰的に次のように定義される。項項 t の $\mathcal{I}$ における値 $t_{\mathcal{I}}$ は次のように定義される。

- t が $f(t_1, \cdots, t_n)$ の形の項の時、

$$t_{\mathcal{I}} = f_{I_1}(t_{1\mathcal{I}}, \cdots, t_{n\mathcal{I}})$$

- t が $\mathbf{X}f(t_1, \cdots, t_n)$ の形の項の時、

$$t_{\mathcal{I}} = f_{I_2}(t_{1\mathcal{I}}, \cdots, t_{n\mathcal{I}})$$

論理式

- A が $P(t_1, \cdots, t_n)$ で P が $\doteq$ でない時、$A_{\mathcal{I}} = P_{I_1}(t_{1\mathcal{I}}, t_{2\mathcal{I}}, \cdots, t_{n\mathcal{I}})$.
- A が $t \doteq u$ の時、$t_{\mathcal{I}} = u_{\mathcal{I}}$ ならば *true*, そうでなければ *false*.
- A が $B \vee C$ および $\neg B$ の時、$A_{\mathcal{I}}$ はそれぞれ $B_{\mathcal{I}}$ と $C_{\mathcal{I}}$ の論理積および $B_{\mathcal{I}}$ の否定。
- $\mathbf{X}A$ に対して $\mathbf{X}A_{\mathcal{I}} = A_{\mathcal{I}^2}$.
- $A\,\mathcal{U}\,B$ に対して、ある $k \geq 0$ が存在して、k 未満の全ての $j \geq 0$ について $A_{\mathcal{I}^j} = true$ かつ $B_{\mathcal{I}^k} = true$ ならば $A\,\mathcal{U}\,B_{\mathcal{I}} = true$. それ以外ならば $A\,\mathcal{U}\,B_{\mathcal{I}} = false$.

論理式 $A_{\mathcal{I}}$ が *true* となることを A が解釈 $\mathcal{I}$ のもとで満足されると言い、$\mathcal{I} \models A$ と書く。論理式 A に対してどのような $\mathcal{I}$ のもとでも $A_{\mathcal{I}} = true$ ならば A は恒真であるといい、$\models A$ と書く。またある $\mathcal{I}$ に対して $A_{\mathcal{I}} = true$ ならば充足可能。そうでないなら充足不能という。 □

3.2 充足可能性判定アルゴリズム

与えられた式 A の恒真性を判定するためには、$\neg A$ の充足不能性を示せば良い。すなわち、$\neg A$ を満足するような $\mathcal{I}$ の有無を示せばよい。$\neg A$ が充足可能の時、またその時に限り、$\mathcal{I}$ に対応する無限の経路を含むようなタブロー (tableau) と呼ばれる有向グラフ $\mathcal{T} = (S_T, R_T)$ を構築する。

定義 7 初等式 (elementary formula) :
与えられた論理式 A について、$P(t_1, \cdots, t_n)$ (P は述語) を $P(t_1, \cdots, t_n) \doteq T$ で置き換えて、P を関数記号と見なす。初等式の集合 $El(A)$ は次のように el を用いて再帰的に定義される。

- $el(t \doteq u) = \{t \doteq u\}$
- $el(\neg g) = el(g)$
- $el(g \vee h) = el(g) \cup el(h)$
- $el(\mathbf{X}g) = \{\mathbf{X}g\} \cup el(g)$
- $el(g\,\mathcal{U}\,h) = \{\mathbf{X}(g\,\mathcal{U}\,h)\} \cup el(g) \cup el(h)$

$El(A)$ は次のように定義される。

- $El(A) = \{t \doteq u \mid t \text{ と } u \text{ が } g \text{ 中に出現する部分項}\} \cup el(A)$

□

[例] $A \stackrel{\text{def}}{=} \neg(x \doteq \mathbf{X}y)\,\mathcal{U}\,(x \doteq \mathbf{X}y)$ に対して $El(A) = \{x \doteq \mathbf{X}y, x \doteq y, \mathbf{X}A\}$ となる。 □

タブローの構成
タブローの節点集合 S_T を次の 2 つの条件 1. および 2. を満足するものと定義する。

1. $\sigma \in S_T$ は式の集合であって、全ての $f \in El(A)$ に対して、f か $\neg f$ のいずれか一方が σ に属する。

2. 全ての $\sigma \in S_T$ に対して、$\mathbf{X}f$ またはその否定の形の式を取り除いたものを σ' とする。項に出現する $\mathbf{X}$ については、1 引数の関数記号とみなす。このとき、$\bigwedge_{g\in\sigma'} g$ が充足可能。

論理式の長さを n とした時、タブローの節点の数は $O(2^{n^2})$ となる。

こうして構成された節点 σ が $\bigwedge_{g\in\sigma} g$ を満足する無限経路の最初の節点となるように、節点間の枝を定める。次の集合 $sat(g)$ を導入する。$sat(g)$ は直観的には、g を満足する経路の最初の節点の集合を表わす。

- $sat(g) = \{\sigma \mid g \in \sigma\}$ where $g \in El(f)$.
- $sat(\neg g) = \{\sigma \mid \sigma \notin sat(g)\}$.
- $sat(g \vee h) = sat(g) \cup sat(h)$.
- $sat(g\ \mathcal{U}\ h) = sat(h) \cup (sat(g) \cap sat(\mathbf{X}(g\ \mathcal{U}\ h)))$.

定義 8 論理式 A を $El(g)$ の要素を原子命題とみなして、積和標準形に変換したものを完全な積和形と呼ぶ。 □

積項 C_i は一般に $\bigwedge_i c_i$ の形をしているが、以下では、$\bigcup_i\{c_i\}$ と同一視する。

性質 1 与えられた式 g を完全な積和形 $\bigwedge_i C_i$ に変形する。$sat(g)$ の任意の要素はある C_i と一致する。 □

タブローの枝の集合 R_T を次のように定める。
$(\sigma, \sigma') \in R_T$ となるのは次の 2 条件が同時に満足される時その時に限る。

- $\bigwedge_{\mathbf{X}g\in El(f)} (\sigma \in sat(\mathbf{X}g) \Leftrightarrow \sigma' \in sat(g))$
- σ' 内の全ての動的関数 f を $\mathbf{X}f$ で置き換えたものを σ'' とする。このとき $\sigma \cup \sigma''$ のうち、$t \doteq u$ の形をした式およびその否定の式のみからなる集合が充足可能。

タブローの枝には次の性質がある。

性質 2 1. $(\sigma, \sigma') \in R_T$ ならば、σ に含まれる $\mathbf{X}g_i$ $(i = 1, \cdots, n)$ と $\neg\mathbf{X}h_j$ $(j = 1, \cdots, m)$ の形の式に対して、σ' は $\bigwedge_i g_i \wedge \bigwedge_j h_j$ の完全な積和形の積項である。

2. タブロー上の有限経路 $\sigma_0, \sigma_1, \cdots, \sigma_n$ について、σ_i 中の動的関数記号 f を $\mathbf{X}^{(i)}f$ で置き換える。ここで $\mathbf{X}^{(i)}$ は $\mathbf{X}$ を i 個並べたもの。このとき $\bigcup_i \sigma_i$ は充足可能。

上記で構成したタブローが無限経路を含んでいれば、A は充足可能であるといいたい。しかし、このタブローでは $\sigma \in sat(g\ \mathcal{U}\ h)$ が存在すると、g は成立しつづけるが h が永久に成立しないような経路が存在し得る。このような経路でない無限経路を見つけることは $sat(g\ \mathcal{U}\ h \wedge \neg h)$ のみの節点からなる経路以外の無限経路を見つけることに等しく、これは、すなわち次の条件を満足する経路が見つけることを意味する。

条件 1 $sat(\neg(g\ \mathcal{U}\ h \wedge \neg h))$ の節点のいずれかを無限にしばしば通過する。

結局、タブローの強連結成分を求めて、$sat(\neg(g\ \mathcal{U}\ h \wedge \neg h))$ の要素を含む強連結成分のうち、与えられた論理式 A について $sat(A)$ の状態のいずれかから到達できるものが存在することと A が充足可能であることが等価となる。

定理 1 論理式 A が充足可能であるのは、A から導出されるタブロー上の節点 $\sigma \in sat(A)$ を始点とする無限経路で条件 1を満足するものが存在する時、またその時に限る。 □

(証明) 節点 $\sigma \in sat(A)$ を始点とする無限経路で条件1を満足するものが存在すれば A が充足可能であることを示す。このような無限経路の一つを ρ として、これから A が充足される解釈 $\mathcal{I}$ を構成すれば良い。$\rho = \sigma_1, \sigma_2, \cdots$. ただし、$\sigma \in S_T$ とする。

まず、$\mathbf{X}f$ と $\neg\mathbf{X}f$ の形の式を無視して、関数記号(述語記号 P も含む)に対する解釈を定める。σ_i には、$\mathbf{X}t$ の形式の項が出現するが、$\mathbf{X}$ を動的関数記号とみなして、$\mathbf{X}t$ の値も考えた解釈を $\mathcal{I}' = I'_1, I'_2, \cdots$ を I'_1 から順に決定する。I'_1 ついては、時相演算子を含まない 2 章の場合と同様に、ρ_1 を充足するように定める。I'_i が決定しているとする。I'_{i+1} は σ_i と σ_{i+1} を、R_T を構成する時の条件 2. で考えたように、併合して、同値となった項については、I'_i における値と同一の値を持つように I'_{i+1} を定める。このように定めると σ_i における Xt の形の項の値は、σ_{i+1} における t の値と一致する。次に、$I'_1, I'_2, \cdots$ を通常の関数記号への値の割り当てだけに制限した $\mathcal{I} = I_1, I_2, \cdots$ を構成する。各 σ_i に含まれる $t \doteq u$ および $\neg r \doteq w$ の形の式は、この解釈のもとで $true$ となる。

次に $\mathbf{X}f$ と $\neg\mathbf{X}f$ を考慮して、上記で構成した $\mathcal{I}$ について、$\sigma_i \in sat(g)$ となることと、$\mathcal{I}^i \models g$ となることが同値であることを式の長さに関する帰納法によって示す。g が $t \doteq u$ の形の式の時には、$t \doteq u \in \sigma_i$ なので上記の議論より明らか。g が $\neg f$ の形の式の時、$\sigma_i \in sat(f)$ と $\mathcal{I}^i \models f$ が同値ならば、$\sigma_i \in sat(\neg f)$ と $\mathcal{I}^i \models \neg f$ も同値であることがいえる。g が $f \vee h$ のときも同様に示すことができる。g が $\mathbf{X}f$ の形の式の場合を考える。R_T の構成の仕方から、$\sigma_i \in sat(\mathbf{X}f)$ と $\sigma_{i+1} \in sat(f)$ が同値である。帰納法の仮定より、$\sigma_{i+1} \in sat(f)$ と $\mathcal{I}^{i+1} \models f$ が同値なので、結局、$\sigma_i \in sat(\mathbf{X}f)$ と $\mathcal{I}^i \models \mathbf{X}f$ は同値。g が $f\ \mathcal{U}\ h$ の形の式の場合を考える。$\sigma_i \in sat(f\ \mathcal{U}\ h)$ となることと、$\sigma_i \in sat(h)$ または $\sigma_i \in sat(f \wedge \mathbf{X}f\ \mathcal{U}\ h)$ は同値。条件1を考えると、$\sigma_i \in sat(f\ \mathcal{U}\ h)$ であることと、ある $k \geq i$ について $\sigma_k \in sat(h)$ かつ $i \leq j < k$ なる j について、$\sigma_j \in sat(f)$ であることが同値。帰納法の仮定より、$\sigma_i \in sat(f\ \mathcal{U}\ h)$ であることと $\mathcal{I} \models f\ \mathcal{U}\ h$ が同値となる。以上より、節点 $\sigma \in sat(A)$ を始点とする無限経路で条件1を満足するものが存在すれば A が充足可能であることが言えた。

逆に、論理式 A が充足可能であるとき、条件1を満足する無限経路がタブロー上に存在することを示す。A を満足するような解釈を $\mathcal{I}$ とする。A を完全な積和形に変形したとき、少なくとも一つの積項 C_1 が存在して、$\mathcal{I} \models C_1$ となる。C_1 のうち、$\mathbf{X}g_i$ と $\neg\mathbf{X}h_j$ の形の式に対して、$\bigwedge_i g_i \wedge \bigwedge_j h_j$ の完全な積和形を考えると、少なくとも一つの積項 C_2 について、$\mathcal{I} \models C_2$ となる。同様にして、$\mathcal{C} = C_1, C_2, C_3, \cdots$ なる積項の無限列が $\mathcal{I}$ に対して定義できる。C_i と C_{i+1} は R_T の条件 1. および 2. を満足する。したがって、$\mathcal{C}$ はタブロー上に存在する無限経路である。以上で定理が証明された。 □

謝辞

有益なご助言、ご討論頂いた京都大学情報工学科矢島研究室の皆様に感謝いたします。

参考文献

[1] R. E. Bryant. Graph-Based Algorithms for Boolean Function Manipulation. *IEEE Transactions on Computers*, C-35(8):677–691, August 1986.

[2] G. Nelson and D. C. Oppen. Simplification by Cooperating Decision Procedure. *ACM Trans. on Programming Languages and Systems*, 1(2):245–257, 1979.

[3] R. B. Jones, D. L. Dill, and J. R. Burch. Efficient Validity Checking for Processor Verification. In *Proceedings of International Conference on Computer-Aided Design*, pages 2–6, November 1995.

[4] N. Rescher and A. Urquhart. *Temporal Logic*. Springer-Verlag, 1971.

[5] E. Clarke, O. Grumberg, and K. Hamaguchi. Another Look at LTL Model Checking. In *Conference on Computer-Aided Verification, LNCS 818*, pages pp.415–427, June 1994.

[6] J. H. Gallier. *Logic for Computer Science*. John Wiley & Sons, 1987.

[7] G. Nelson and D. C. Oppen. Fast Decision Proceudres Based on Congruence Closure. *J. ACM*, 27(2):356–264, 1980.